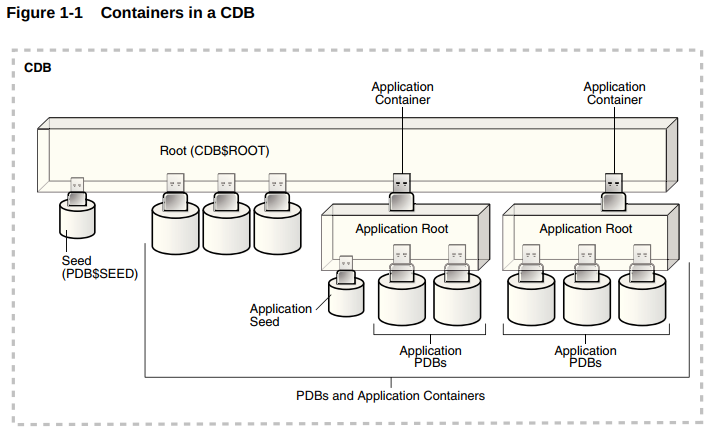
Oracle Multitenant Admin Guide 19c

# Multitenant Architecture

## Introduction to the Multitenant Architecture



Every CDB has the following containers:

* Exactly one CDB root container (also called simply the root). The CDB root is a collection of schemas, schema objects, and nonschema objects to which all PDBs belong The root stores Oracle-supplied metadata and common users. An example of metadata is the source code for Oracle-supplied PL/SQL packages. A common user is a database user known in every container. The root container is named CDB$ROOT.
* Exactly one system container. The system container includes the root CDB and all PDBs in the CDB. Thus, the system container is the logical container for the CDB itself.
* Exactly one seed PDB. The seed PDB is a system-supplied template that the CDB can use to create new PDBs. The seed PDB is named PDB$SEED. You cannot add or modify objects in PDB$SEED.
* Zero or more application containers. An application container consists of exactly one application root, and the PDBs plugged in to this root. Whereas the system container contains the CDB root and all the PDBs within the CDB, an application container includes only the PDBs plugged into the application root. An application root belongs to the CDB root and no other container.
* Zero or more user-created PDBs. A PDB contains the data and code required for a specific set of features. For example, a PDB can support a specific application, such as a human resources or sales application. No PDBs exist at creation of the CDB. You add PDBs based on your business requirements. A PDB belongs to exactly zero or one application container. If a PDB belongs to an application container, then it is an application PDB. Overview of the Multitenant Architecture

## Overview of the Multitenant Architecture

### The CDB Root and System Container

### PDBs

* Standard PDB
* Application Root
* Seed PDBs
* Proxy PDBs (refers to a remote PDB, called the referenced PDB)

### Applications in an Application Container

An application container is an optional, user-created CDB component that stores data and metadata for one or more application back ends. A CDB includes zero or more application containers.

For example, you might create multiple sales-related PDBs within one application container, with these PDBs sharing an application back end that consists of a set of common tables and table definitions. You might store multiple HR-related PDBs within a separate application container, with their own common tables and table definitions

### Tablespaces and Database Files in a CDB

A CDB contains the following files:

* One control file
* One online redo log
* One or more undo tablespaces
  + Local undo mode ( default)
  + Shared undo mode
* SYSTEM and SYSAUX tablespaces for every container
* Zero or more user-created tablespaces
* A set of temp files for every container

# Creating and Configuring a Multitenant Environment

## Overview of Configuring and Managing a Multitenant Environment

## Creating and Configuring a CDB

# Creating and Removing PDBs and Application Containers

## Creating and Configuring a CDB

## Creating a PDB from Scratch

## Cloning a PDB or Non-CDB

## Relocating a PDB

## Plugging In an Unplugged PDB

## Creating a PDB as a Proxy PDB

## Removing a PDB

## Creating and Removing Application Containers and Seeds

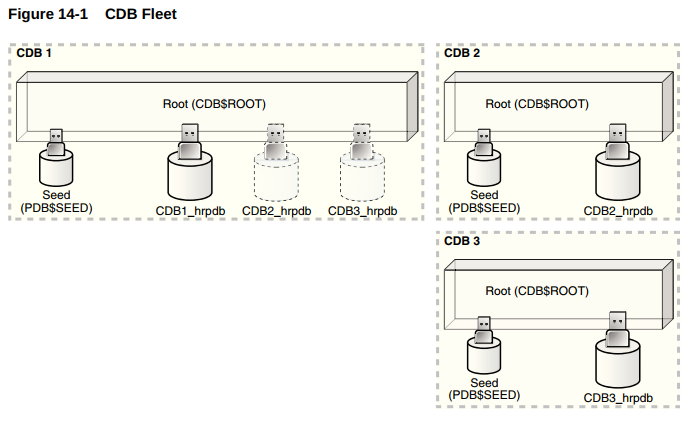
# Administering a Multitenant Environment

## Administering a CDB

## Administering a CDB Fleet

A lead CDB is the central location for monitoring and managing the CDBs in the fleet.Designate one CDB in the fleet to be the lead CDB by setting its LEAD\_CDB database property to TRUE. The other CDBs in the fleet point to the lead CDB by setting the LEAD\_CDB\_URI database property. After you configure the CDB fleet, PDB information from the various CDBs is synchronized with the lead CDB. All PDBs in the CDBs are now “visible” in the lead CDB, enabling you to access the PDBs in the fleet as a single, logical CDB from the lead CDB.

The following figure shows a CDB fleet consisting of CDB1, CDB2, and CDB3. The lead CDB is CDB1. CDB2\_hrpdb, which resides in CDB2, is visible in CDB1. CDB3\_hrpdb, which resides in CDB3, is also visible in CDB1.



## Administering PDBs

## Administering a PDB Snapshot Carousel

A PDB snapshot carousel is a library of PDB snapshots. A PDB snapshot is a point-in-time copy of a PDB. You can create snapshots manually using the SNAPSHOT clause of CREATE PLUGGABLE DATABASE (or ALTER PLUGGABLE DATABASE), or automatically using the EVERY interval clause.

Starting in Oracle Database 19c, you can make snapshots of source PDBs in read/write mode. The content of a PDB snapshot depends on the setting of the CLONEDB initialization parameter when the snapshot is created. When CLONEDB is TRUE, the snapshot is a sparse copy of the PDB data files. When CLONEDB is FALSE, the snapshotis a full copy of the PDB files.

## Administering Application Containers

## Managing Security for a Multitenant Environment

## Monitoring CDBs and PDBs

# Using Oracle Features in a Multitenant Environment

## Backing Up and Recovering CDBs and PDBs

## Using Database Utilities in a Multitenant Environment

## Using Oracle Resource Manager for PDBs

## Using Oracle Scheduler with a CDB

## Using Oracle Database Vault with a CDB

## Using XStream with a CDB